

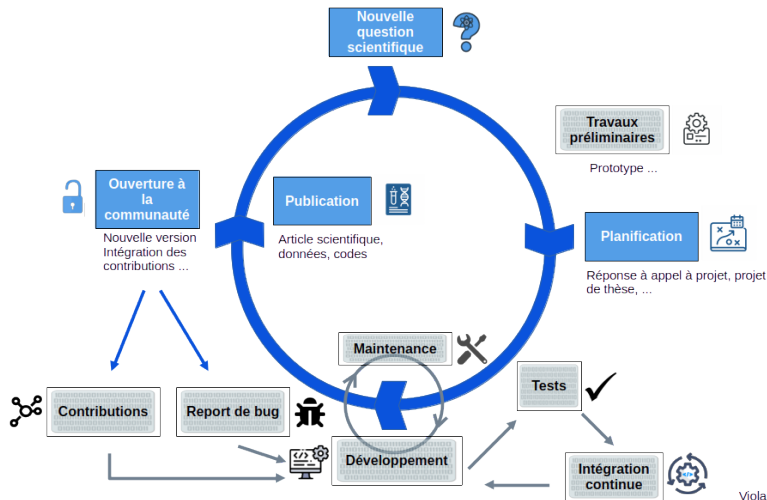
SWHID : identifiant intrinsèque universel pour les logiciels

Violaine Louvet

Séminaire du Consortium DataCite France, novembre 2025



Le logiciel de recherche



Le logiciel dans la science ouverte

Omniprésent dans la recherche

- **Elément essentiel** des processus de recherche dans la **quasi totalité des disciplines scientifiques**
- Approche qui peut être différente d'une communauté à l'autre selon que le logiciel est un **outil, un résultat ou un objet d'études**

Intrinsèquement ancré dans l'ouverture

- La plupart des développements réalisés dans les laboratoires de recherche le sont sous **licence libre**
- Le logiciel libre est né dans la pratique universitaire, **précurseur du mouvement de la science ouverte**
- Mais il n'est **intégré que très récemment** dans les politiques de science ouverte
- Et **en dehors** des « circuits » habituels de publications, citations ...

Le logiciel n'est pas une publication comme les autres

Publications

- Statiques
- Linéaires
- Granularité simple
- Version finale figée



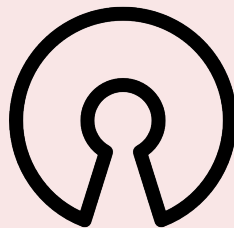
Données

- Relativement stables
- Structurées (tableaux, bdd)
- Métadonnées standards
- Version unique



Logiciels

- **Dynamiques**
- **Modulaires**
- **Versionnés**
- **Granularité multiple**



Les défis spécifiques du logiciel

Granularité multiple

Un logiciel **n'est pas un objet monolithique** :

- Le projet complet
- Une version spécifique (release)
- Un commit particulier
- Un répertoire
- Un fichier précis
- Même une ligne de code !

Besoin pour la reproductibilité

On doit pouvoir retrouver la version **EXACTE** du code utilisé dans une expérience, y compris dépendances et environnement, pas juste « version 2.0 ».

La citation du logiciel

Dans l'idéal

Pour la citation, on doit pouvoir référencer :

- Non seulement le **logiciel complet**
- Mais aussi une **fonction spécifique** réutilisée
- Ou un **algorithme particulier** dans un fichier ...

Mais la plupart du temps

On donne l'**url du dépôt** du logiciel sur la forge sur laquelle il est développé, ou le **site web** du chercheur où on peut télécharger l'archive des sources ...

The code used to produce the numerical experiments in this article can be found on the Github repository <https://github.com/maxhirsch/Multiscale-SGDCT> [19].

For the reader's convenience, all numerical codes are made available on the web page <http://nabuco.math.cnrs.fr>.

Pourquoi ce n'est pas satisfaisant ?

- La mise à disposition d'un code ayant été utilisé pour un article est un pré-requis à la reproductibilité et à l'éthique scientifique. **C'est une excellente chose !**
- Mais le faire via le dépôt d'une forge, ou un site web présente des risques pour la **pérennisation de l'accès au code**
- Car ce sont des ressources qui **peuvent disparaître**
 - Le chercheur change de laboratoire et déplace sa page web
 - La forge est fermée : par exemple Google Code et Gitorious ...
- Mais aussi car le code a **évolué sur le dépôt** depuis la publication



Software Heritage : la « bibliothèque d'Alexandrie » du code

- Infrastructure portée par l'Inria et l'Unesco
- Dédiée à l'archivage du « patrimoine logiciel mondial de l'humanité »
- Trois missions de Software Heritage : collecte, préservation, partage du code source des logiciels rendus publics.

Quelques éléments chiffrés

- Plus de 26 milliards de fichiers sources
- Plus de 400 millions de projets
- Codes provenant de toutes les forges majeures (GitHub, GitLab, Bitbucket...) qui sont moissonnées automatiquement



Software Heritage

Le SWHID : un identifiant adapté au logiciel

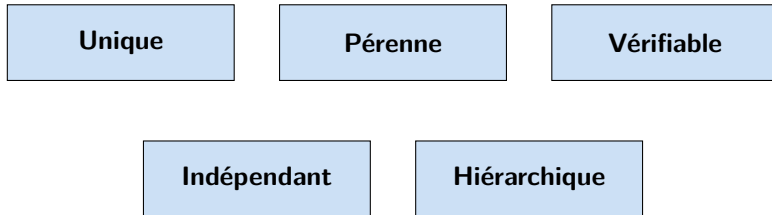
Qu'est-ce qu'un SWHID ?

- SWHID = **Software Hash Identifier**
- Principe fondamental : **identifiant intrinsèque**

Identifiant extrinsèque vs intrinsèque

- **Identifiant extrinsèque** : par exemple, DOI
 - Attribué par une autorité centralisée
 - Nécessite un enregistrement
 - A généralement un coût
- **Identifiant intrinsèque** : par exemple SWHID
 - Calculé à partir du contenu même du code
 - Aucune autorité centrale requise
 - Gratuit et instantané, vérifiable par n'importe qui

Propriétés du SWHID

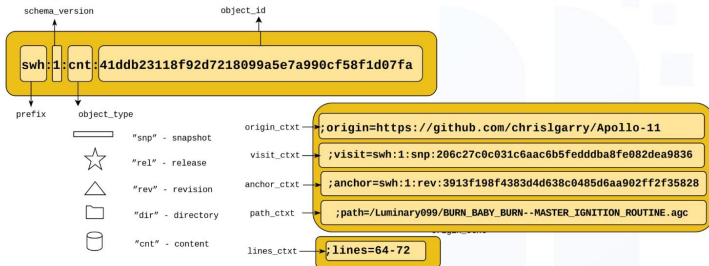


- **Unique** : deux codes différents auront forcément deux SWHID distincts
- **Pérenne** : ne change jamais
- **Vérifiable** : recalculable pour vérifier l'authenticité
- **Indépendant** : pas de plateforme ou organisation
- **Hiérarchique** : différents niveaux de granularité

Structure d'un SWHID

Format général

`swh:1:type:hash_sha1;qualificatifs`



Les 5 types d'objets

Type	Code	Description	Exemple d'usage
Content	cnt	Un fichier individuel	Citer une fonction précise
Directory	dir	Un répertoire	Référencer un module
Revision	rev	Un commit Git	État exact à un instant donné
Release	rel	Une version étiquetée	Version 2.3.0 d'un logiciel
Snapshot	snp	État complet du dépôt	Toutes les branches

Les qualificatifs (optionnels)

Ajoutent du contexte

```
swh:1:cnt:4d99d2d1...;  
  origin=https://github.com/projet/code;  
  visit=swh:1:snp:d7f1b9eb...;  
  anchor=swh:1:rev:2db18992...;  
  path=/src/module.py;  
  lines=42-58
```

Qualificatifs de contexte

- `origin` : URL du dépôt
- `visit` : moment de l'archivage
- `anchor` : point de départ
- `path` : chemin vers le fichier

Qualificatif de fragment

- `lines` : lignes précises
- Ex : `lines=42-58`

SWHID : standard international ISO/IEC (18670:2025)



The image shows the front cover of the ISO/IEC 18670:2025 standard. It features the ISO and IEC logos at the top left, followed by 'International Standard' and 'ISO/IEC 18670:2025'. The main title 'Information technology — SoftWare Hash Identifier (SWHID) Specification V1.2' is prominently displayed, along with 'Edition 1' and '1911-16'.

ISO/IEC 18670:2025

Information technology — SoftWare Hash Identifier (SWHID) Specification V1.2

Published (Edition 1, 2025)

[Read sample](#)

Le SWHID est reconnu depuis avril 2025 comme standard ouvert, un moyen fiable et universel de référencer les constituants logiciels

En pratique, quel SWHID choisir ?

Je suis doctorant et je veux citer le code de ma thèse

→ Utilisez une **release** : version stable et nommée de votre logiciel

```
swh:1:rel:22ece559cc7cc2364edc5e5593d63ae8bd229f9f
```

Ma publication porte sur un algorithme précis

→ Ciblez les **lignes de code** exactes avec le qualifier `lines`

```
swh:1:cnt:4d99d2d18326621ccee996392c2f14ad13a4c3e  
;lines=142-187
```

Je veux que mes résultats soient reproductibles

→ Citez le **commit** exact utilisé pour générer vos résultats

```
swh:1:rev:309cf2674ee7a0749978cf8265ab91a60aea0f7d
```

Format : `swh:1:<type>:<hash>[;qualifier=valeur]`

Complémentarité des identifiants

ID	Rôle	Force	Usage
DOI	Citation académique formelle	Reconnaissance institutionnelle, métadonnées riches	Publier une version "officielle"
HAL-ID	Référencement institutionnel	Intégration bibliographie FR, modération	Valoriser dans contexte institutionnel
SWHID	Accès au contenu exact	Pérennité, granularité, vérifiabilité	Garantir reproductibilité scientifique

Intégration du SWHID dans HAL

Collaboration depuis 2018

CCSD / Inria / Software Heritage

Mode 1 : Dépôt avec code

- Upload du code dans HAL
- Transfert automatique vers SWH
- SWHID généré et intégré

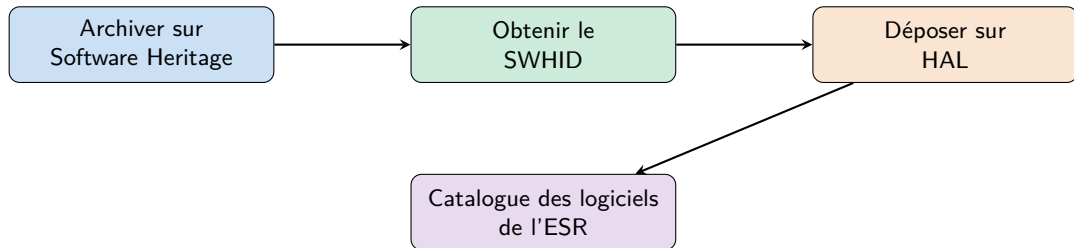
Mode 2 : Dépôt avec SWHID

- Code déjà dans SWH
- Dépôt du SWHID + métadonnées
- Extraction `codemeta.json`

Avantage

Curation par les équipes de modération HAL

Workflow : de Software Heritage au catalogue des logiciels de l'ESR



Pourquoi ce workflow ?

- Pérennité : archivage sur 50+ ans
- Citabilité : identifiant unique
- Visibilité : référencement national

Le catalogue ESR

- Recense les logiciels de la recherche française
- Alimenté automatiquement via HAL
- Valorise les productions logicielles

Conclusions

1 Le logiciel n'est pas un document comme les autres

- Objet vivant, versionné, modulaire
- Besoin d'identifiants adaptés

2 Software Heritage = archive pérenne du code

- Mission patrimoniale
- 50+ milliards d'artefacts

3 SWHID = norme ISO depuis avril 2025

- Standard ouvert international
- Identifiant intrinsèque

4 Complémentarité avec DOI et HAL-ID

- Citation + métadonnées + reproductibilité

Références

- <https://www.swhid.org/>
- <https://www.softwareheritage.org/software-hash-identifier-swhid/>
- https://www.softwareheritage.org/software-heritage-faq/#3_Referencing_and_identification
- <https://programminghistorian.org/fr/lecons/preserver-logiciels-recherche>
- <https://www.pid-network.de/en/news/events/pids-fuer-software>
- <https://zenodo.org/records/15750889>
- <https://logiciels.catalogue-esr.fr/>